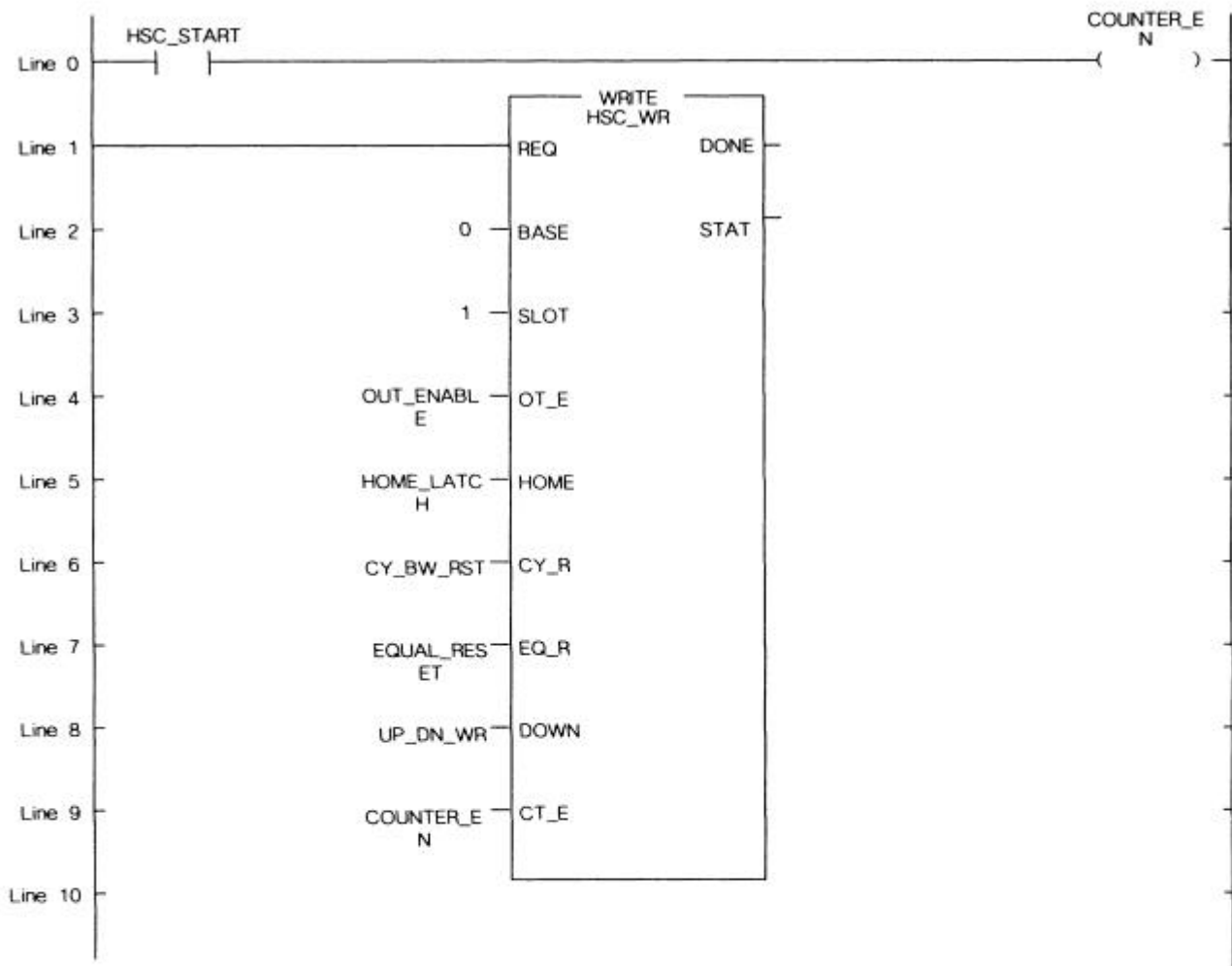# Chapter 5.   GM PROGRAMMING

5.1   Programming Examples

If not especially noted, this section explains programming examples in reference with the G4F – HSCA that is loaded onto the system given below

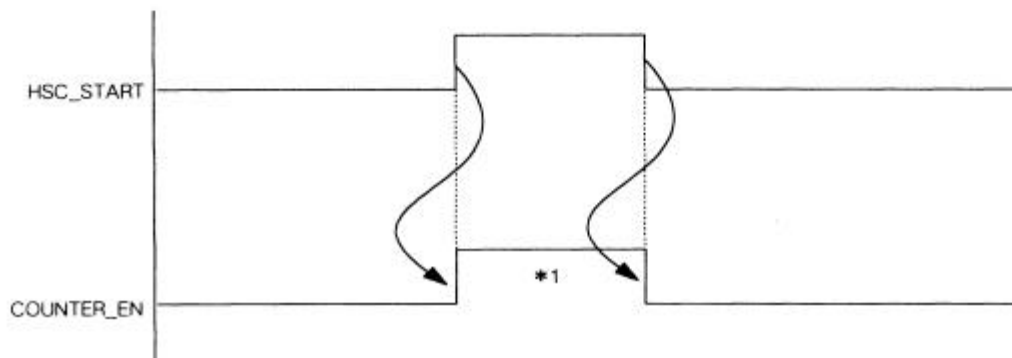### System configuration

GM4-PA2A	GM4-CPUA	G4I-D22A	G4F-HSCA	G4Q-TR2A	

5.1.1   Enabling the count operation

● Timing Diagram

*1. Counting is only possible when the COUNTER_EN is turned on.

5.1.2   Preset
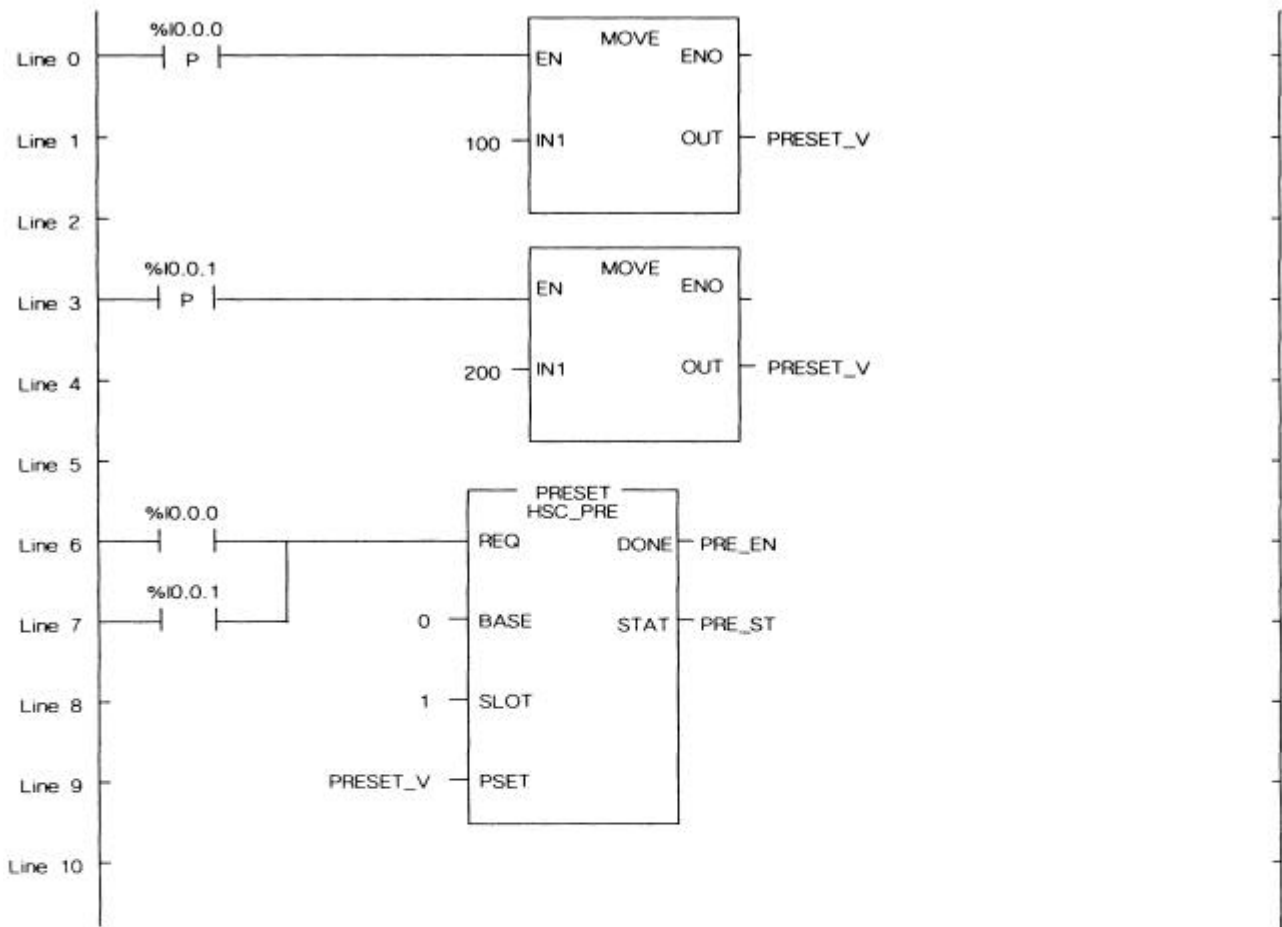
● **Timing diagram**
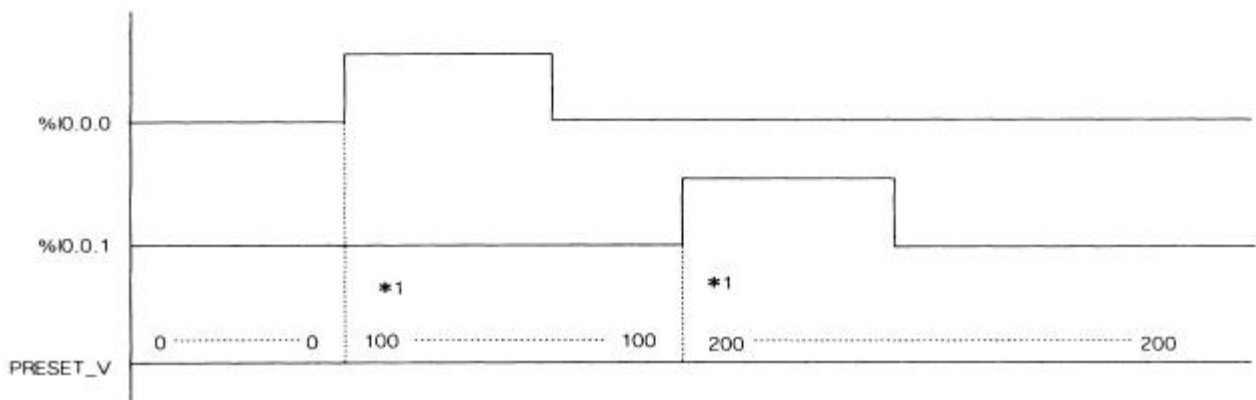
*The function block HSC_PRE will be processed for one scan.

5.1.3   Setting the comparison value
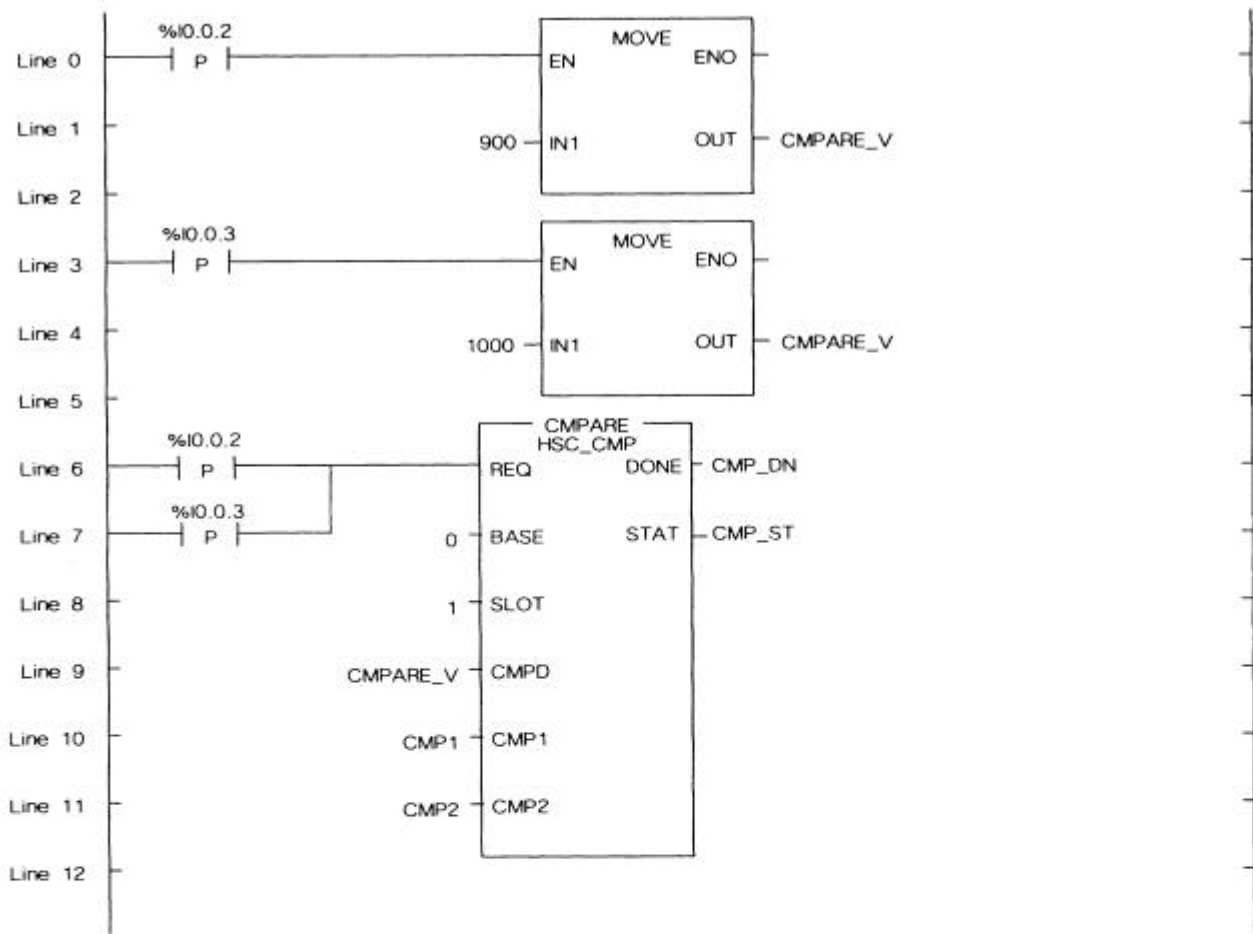
● **Timing Diagram**

*1. The function block HSC_CMP will be processed for one scan.

5.1.4   Setting the magnitude comparison values

● **Timing Diagram**

*1.The function block HSC_CMP will be processed for one scan.

5.1.5   Reading the current count value
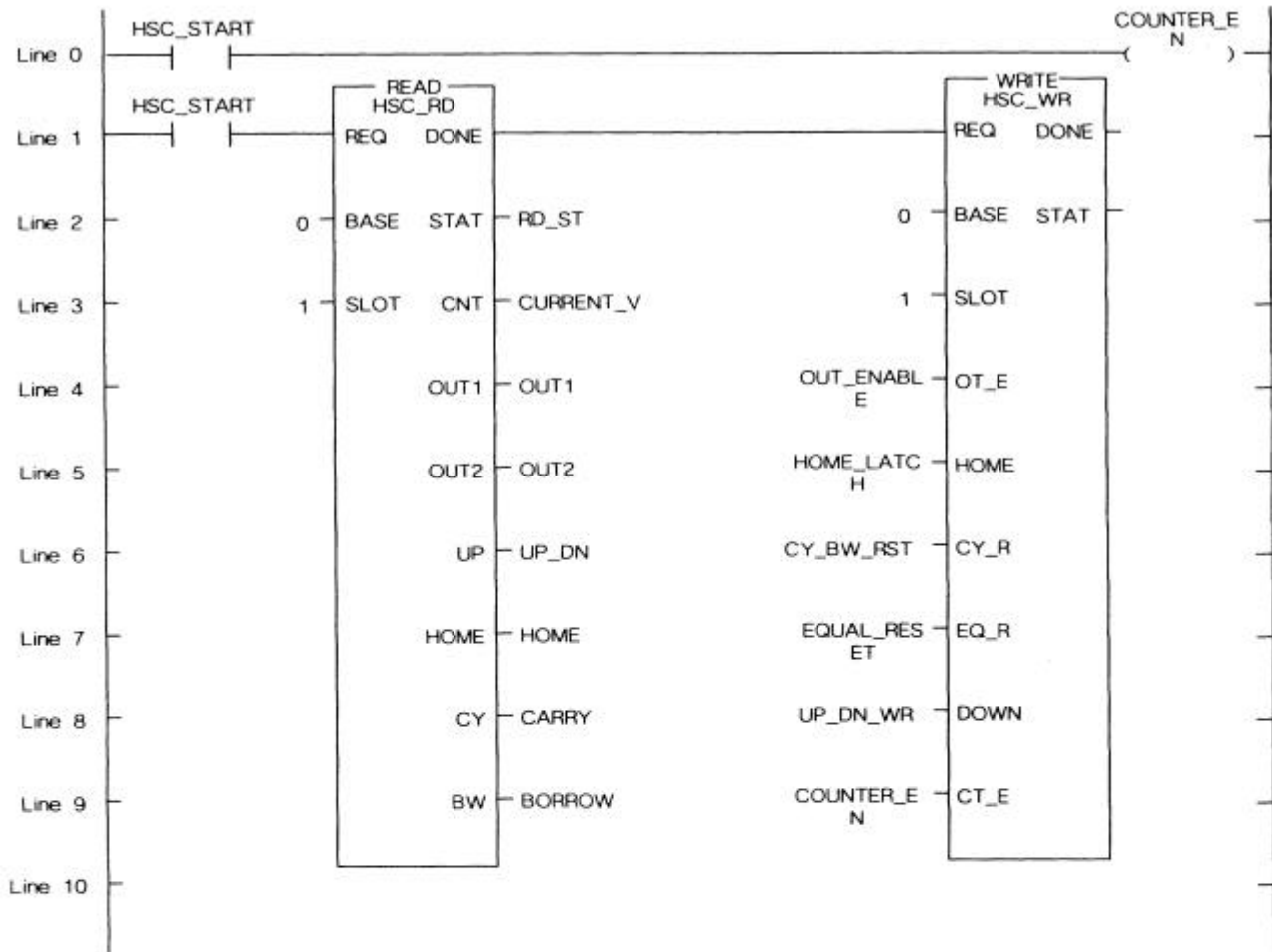
● **Timing Diagram**
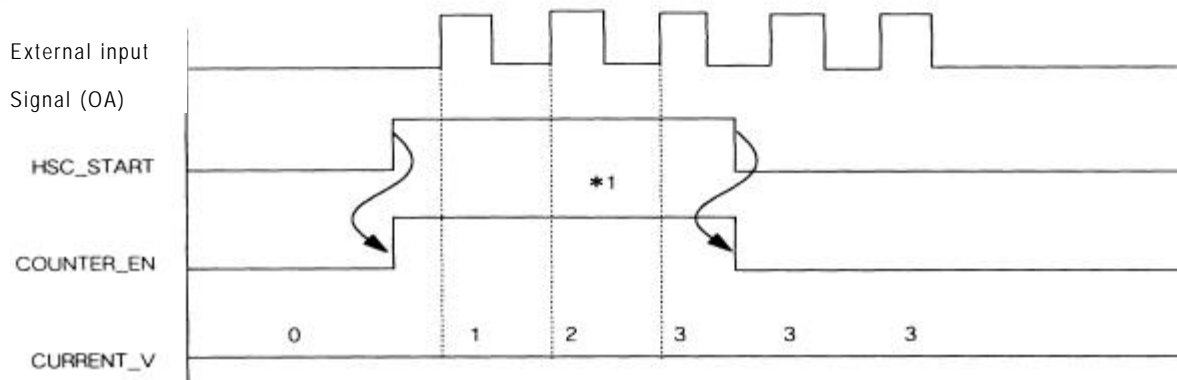
*1.The current count value (CNT) is read only when the COUNTER_EN is turned on.

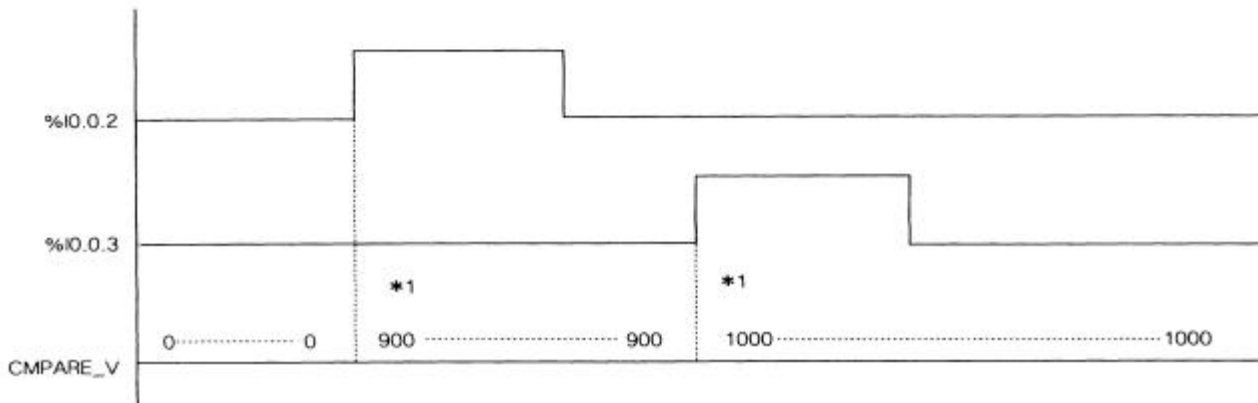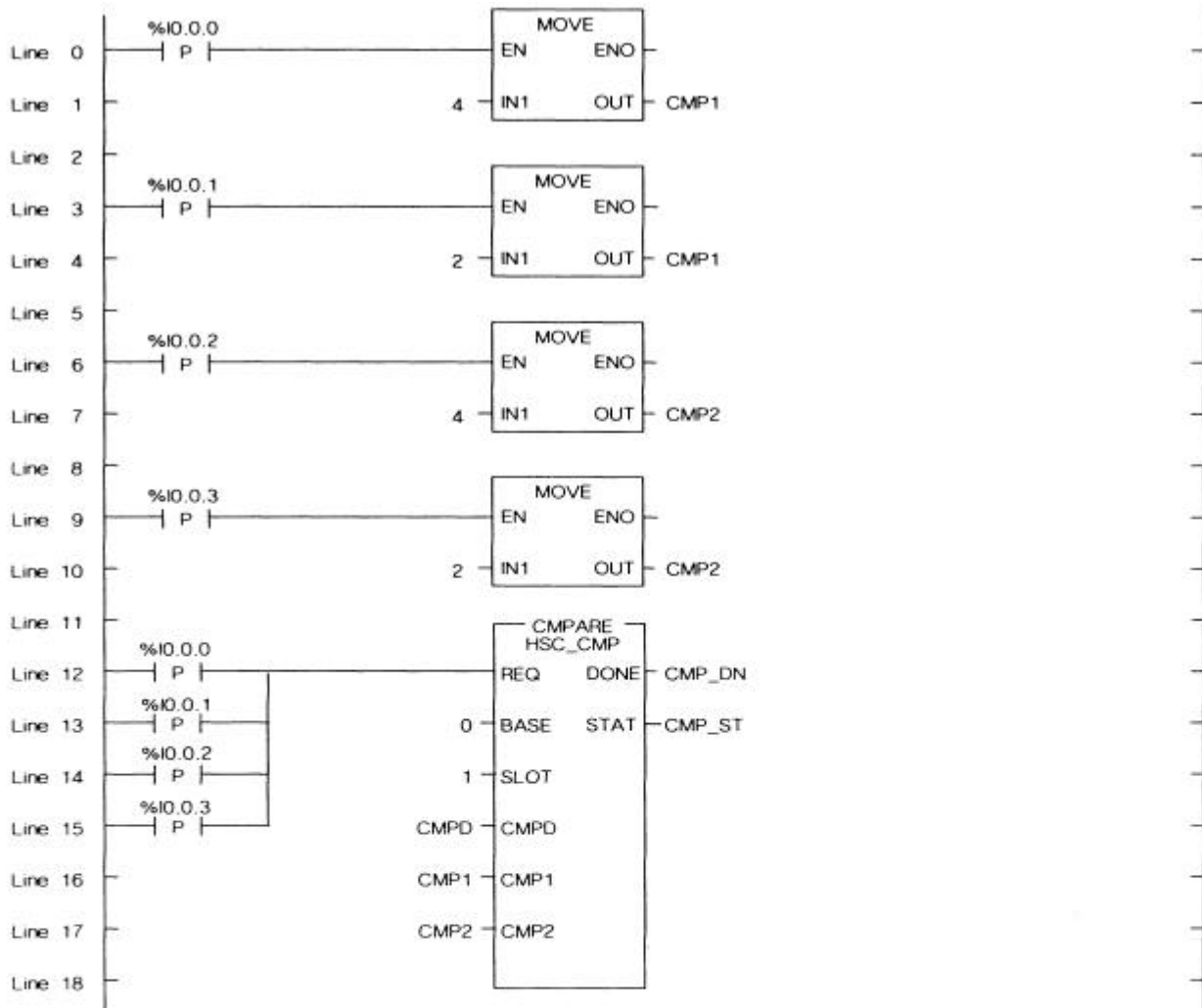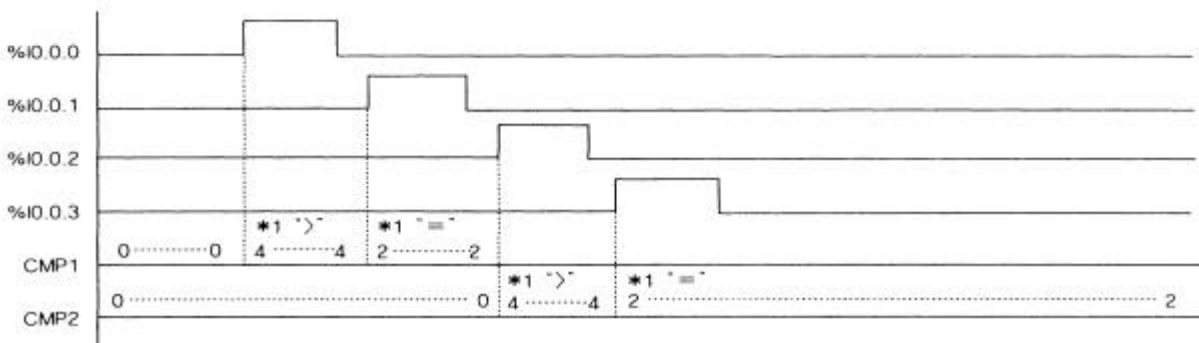## 5.1.6   Enabling the external output
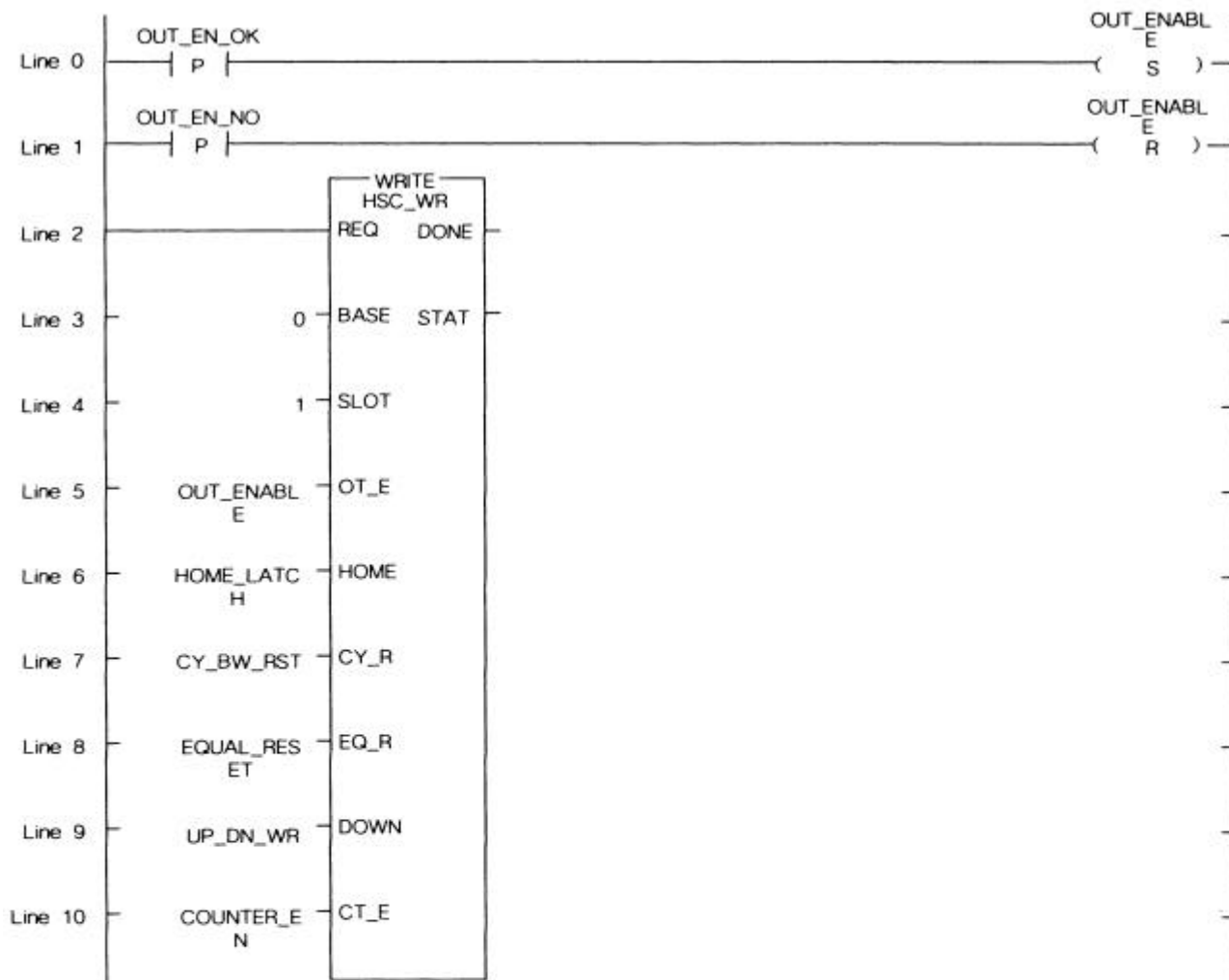
● Timing Diagram

5.1.7 Coincidence reset

```
           EQ_RST_OK                                                EQUAL_RES
                                                                       ET
Line 0  ─────┤ P ├─────────────────────────────────────────────────(     )───

                        ┌─── READ ───┐                    ┌─── WRITE ───┐
                        │   HSC_RD   │                    │   HSC_WR    │
Line 1  ────────────────┤ REQ   DONE ├────────────────────┤ REQ   DONE ├──────

Line 2  ─┤          0 ──┤ BASE  STAT ├─ RD_ST          0 ──┤ BASE  STAT ├──────

Line 3  ─┤          1 ──┤ SLOT   CNT ├─ CURRENT_V      1 ──┤ SLOT        │

Line 4  ─┤              │      OUT1 ├─ OUT1      OUT_ENABL ─┤ OT_E        │
                                                        E

Line 5  ─┤              │      OUT2 ├─ OUT2     · HOME_LATC ─┤ HOME       │
                                                        H

Line 6  ─┤              │        UP ├─ UP_DN     CY_BW_RST ─┤ CY_R        │

Line 7  ─┤              │      HOME ├─ HOME     EQUAL_RES ─┤ EQ_R         │
                                                        ET

Line 8  ─┤              │        CY ├─ CARRY    UP_DN_WR ─┤ DOWN          │

Line 9  ─┤              │        BW ├─ BORROW   COUNTER_E ─┤ CT_E         │
                        └────────────┘            N       └─────────────┘

Line 10 ─┤
```
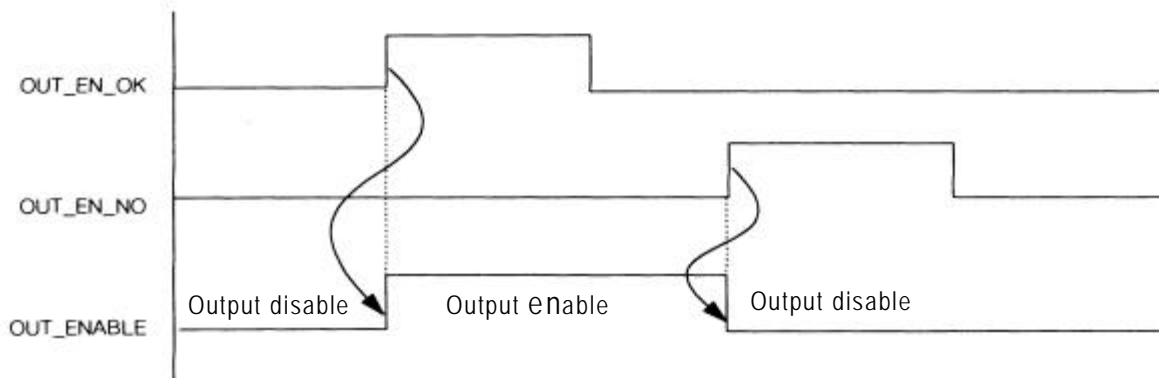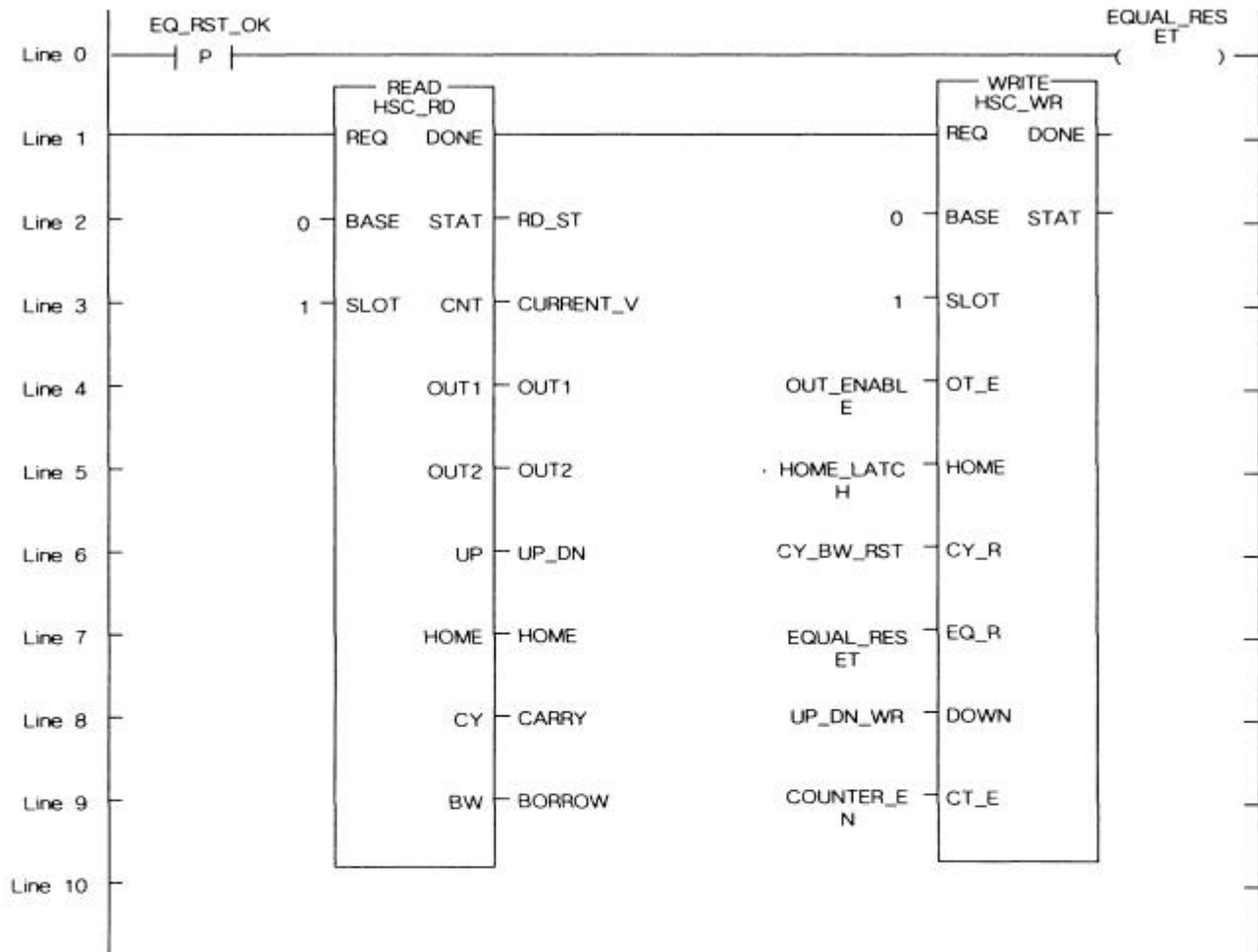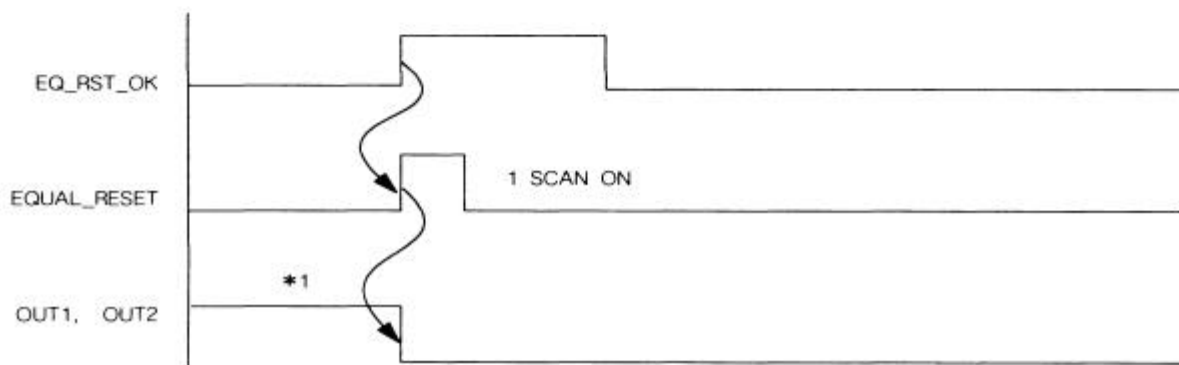
● **Timing Diagram**

*1.This occurs only when the comparison values 1 and 2 are set to " = ", " ≥ ", or " ≤
" and the current count value (CNT) equal to the comparison value (CMPD)

5.1.8 Carry / Borrow reset
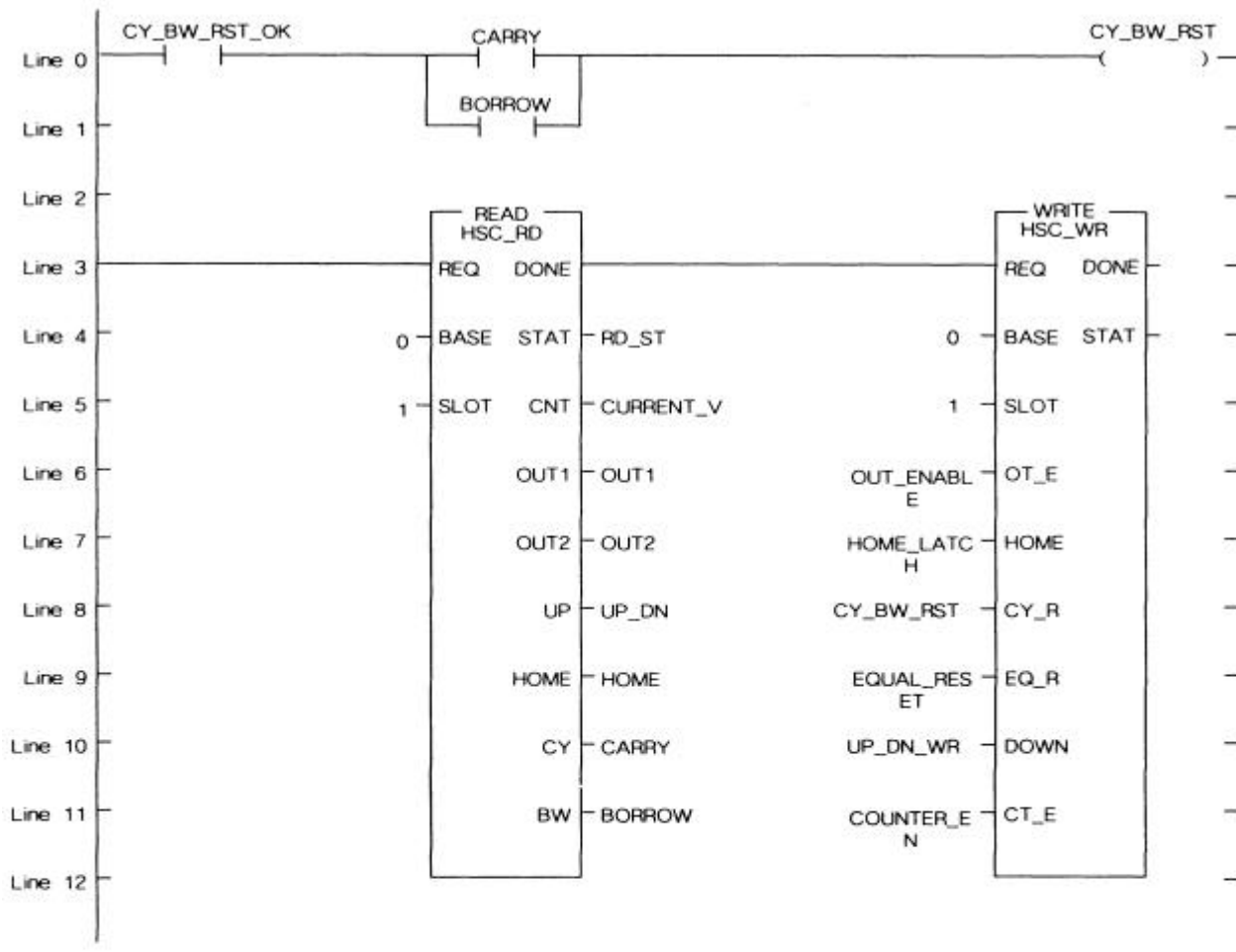
● **Timing Diagram**
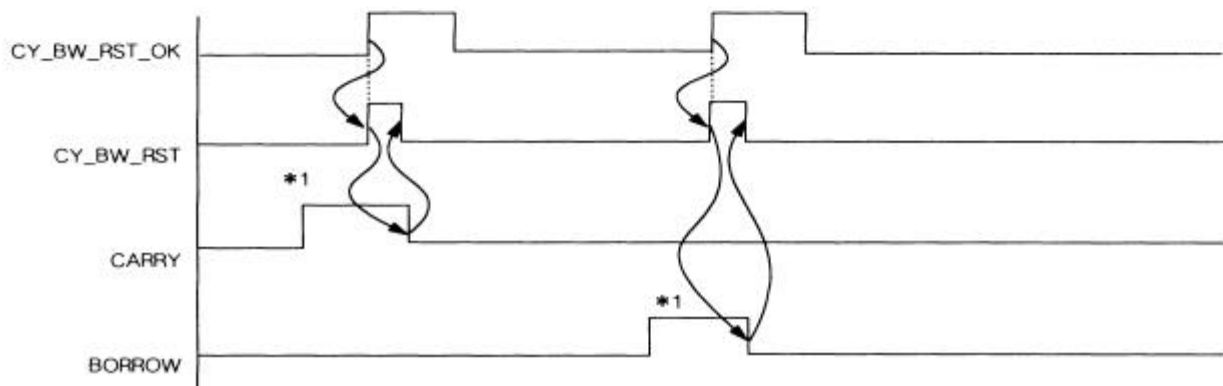
*1. Carry, Borrow signal occurs when the Current count value changes from 16,777,215 to 0
   or from 0 to 16,777,215

5.1.9   Enabling the home latch

Home Latch enable signal is used to set the current count value to 0 when the mechanical reference point has been reached.
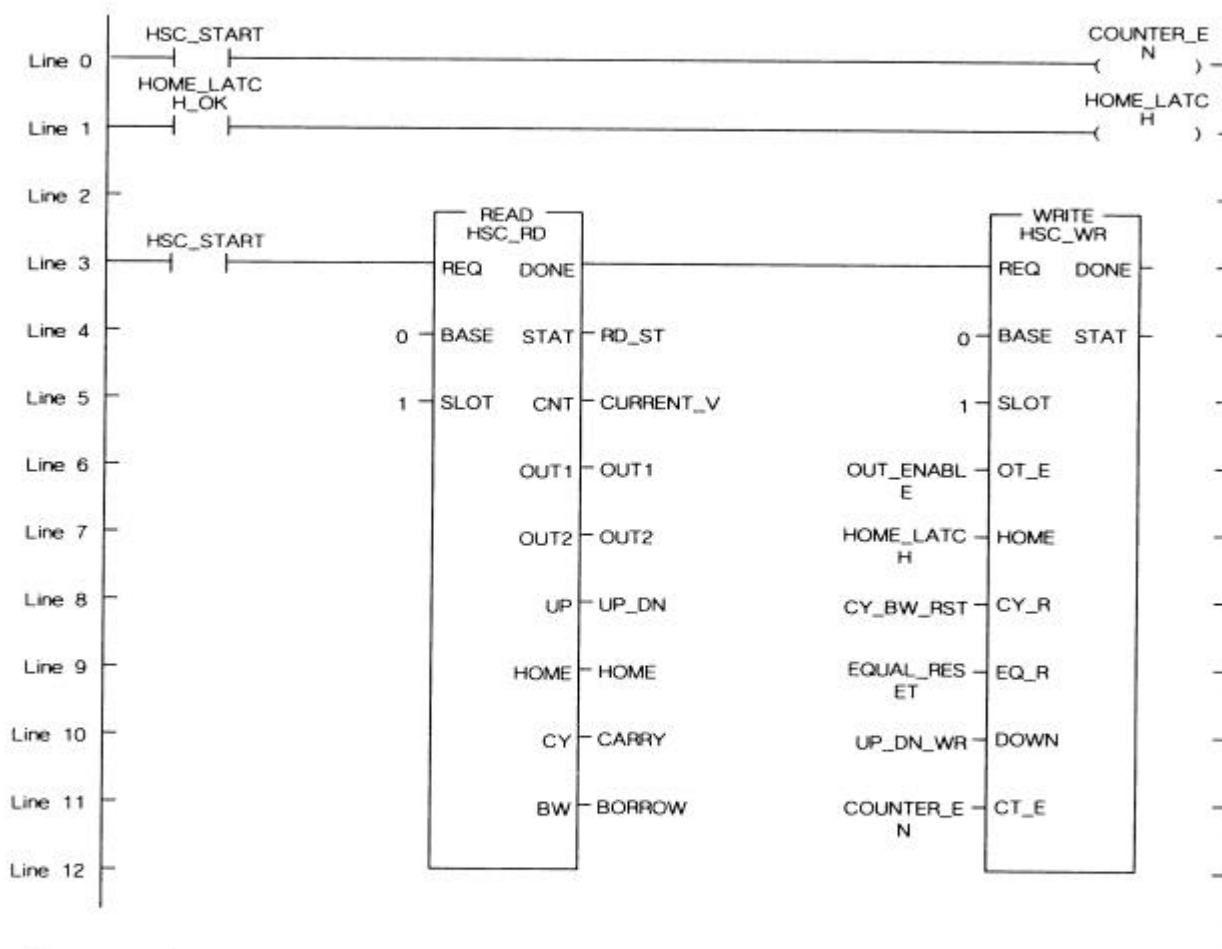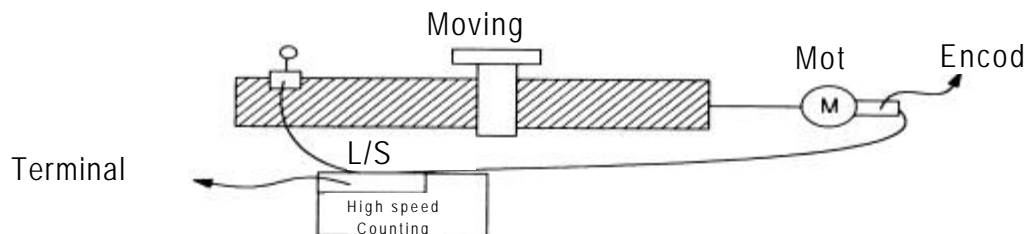
● **Timing diagram**
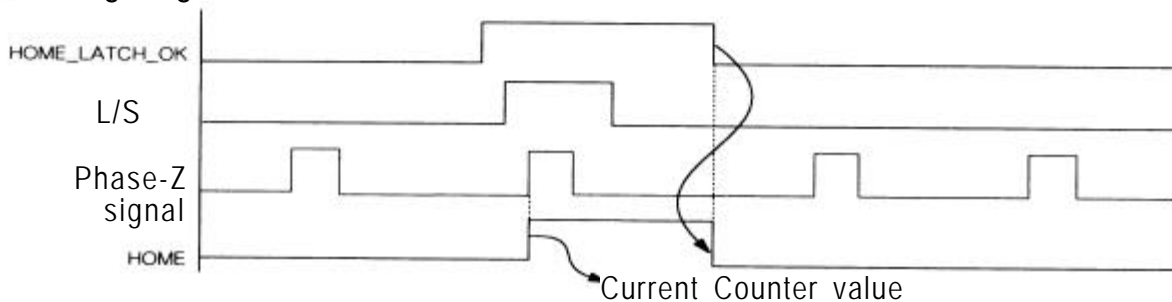
5.1.10   Read/ Write when the high speed counter module Is mounted onto the remote station

System configuration
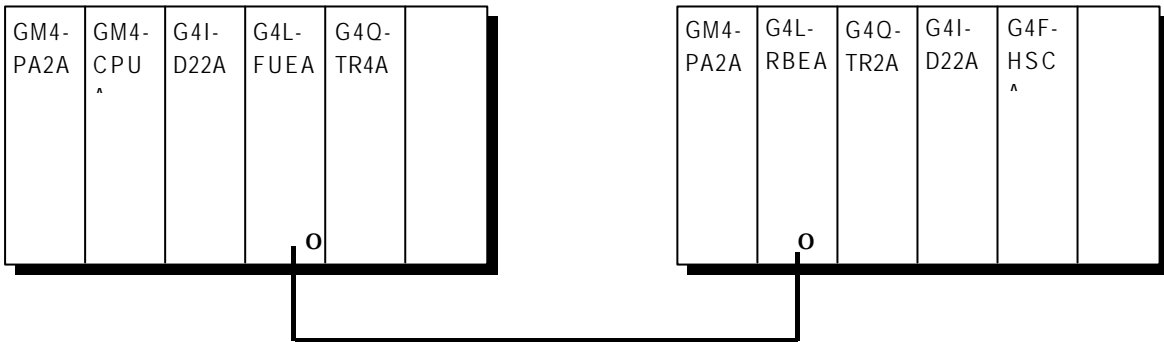
NET-NO : The number of the slot where the G4L – FUEA is loaded = 1

ST-NO :   G4L-RBEA Station No. = 3

BASE :   The number of the base unit where the G4F – HSCA is loaded = 1

SLOT :   The number of the slot where the G4F – HSCA is loaded = 2

Program

```
                                      ┌─ R_WRITE ─┐
           %I0.0.2    WR_NDR          │  HSCROWR  │
Line 17 ├───┤ ├───────┤/├─────────────┤REQ    NDR ├─ WR_NDR
                                      │           │
Line 18 ├                          1 ─┤NET_   ERR ├─   When %I0.0.2 is turned on, write
                                      │ NO        │     is    executed    every    normal
Line 19 ├                          3 ─┤ST_N  STAT ├─    communication   complete   cycle
                                      │  O        │     (NDR)
Line 20 ├                          1 ─┤BASE       │
                                      │           │
Line 21 ├                          2 ─┤SLOT       │
                                      │           │
Line 22 ├                          1 ─┤OT_E       │
                                      │           │
Line 23 ├                    HOME_WR ─┤HOME       │
                                      │           │
Line 24 ├                  CARRY_RST ─┤CY_R       │
                                      │           │
Line 25 ├                     EQ_RST ─┤EQ_R       │
                                      │           │
Line 26 ├                      UP_DN ─┤DOWN       │
                                      │           │
Line 27 ├                          1 ─┤CT_E       │
                                      └───────────┘
Line 28 ├                             ┌─ R_READ ──┐
                                      │  HSCRORD  │
           %I0.0.3   READ_NDR         │           │
Line 29 ├───┤ ├───────┤/├─────────────┤REQ    NDR ├─ READ_NDR
                                      │           │
Line 30 ├                          1 ─┤NET_   ERR ├─   When %I0.0.3 is turned on, read
                                      │ NO        │     is    executed    every    normal
Line 31 ├                          3 ─┤ST_N  STAT ├─    communication   complete   cycle
                                      │  O        │
Line 32 ├                          1 ─┤BASE   CNT ├─ CURRENT_V
                                      │           │
Line 33 ├                          2 ─┤SLOT  OUT1 ├─ OUT1
                                      │           │
Line 34 ├                             │      OUT2 ├─ OUT2
                                      │           │
Line 35 ├                             │        UP ├─ UP_DOWN
                                      │           │
Line 36 ├                             │      HOME ├─ HOME_IN
                                      │           │
Line 37 ├                             │        CY ├─ CY_IN
                                      │           │
Line 38 ├                             │        BW ├─ BW_IN
                                      │           │
Line 39 ├                             └───────────┘
```

● **Timing Diagram**

*1. It is normal communication complete cycle, and it extents in proportion with quantity of

communication data and the number of stations connected to the communication module.

5.2   Application Examples

5.2.1 Program for moving the cart

● System
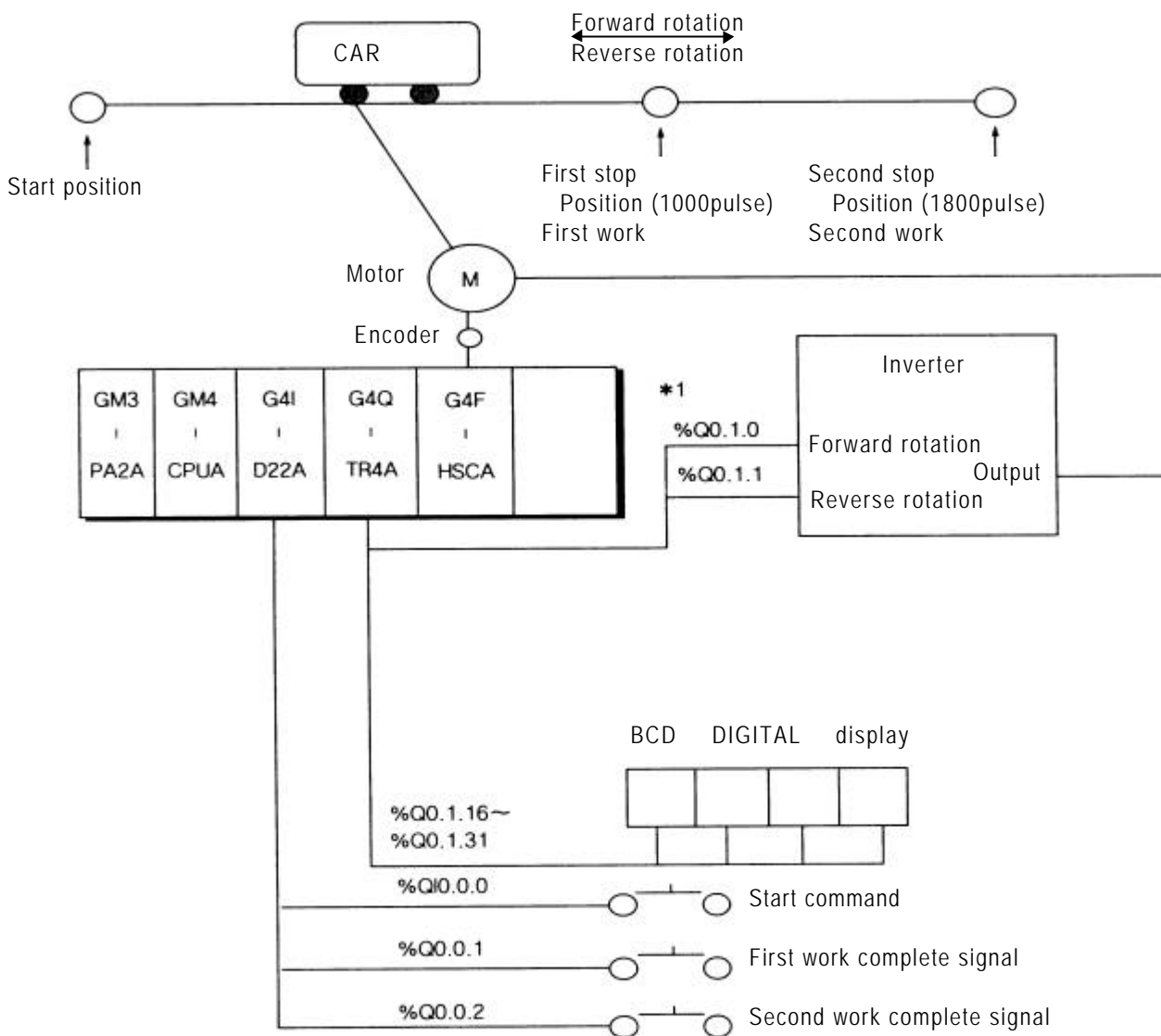
GM4 − CUPA :	GM4 CPU Module
G4I − D22A :	DC Input Module (16
G4Q − TR4A :	TR Output Module (32 Point)
G4F − HSCA :	High-speed counting module

Operation Description

The motor for moving the cart rotates with start command, and makes the cart stop at the first stop position with the High-speed counting module counting the encoder signals from the motor.

Then, if the first work complete signal turns ON, the motor moves the cart to stop at the second stop position. When the second work complete signal turns on, the motor return the cart to the start position.

Input/Output Signal Allocation

%I0.0.0 : Start Command
%I0.0.1 : 1st Work Complete Signal   } Input
%I0.0.2 : 2nd Work Complete Signal

%Q0.1.0 : Motor forward rotation signal (On : forward rotation , Off : Stop)
%Q0.1.1 : Motor reverse rotation Signal (On : Backward rotation, Off : Stop)   } Output
%Q0.1.16 - %Q0.1.31 : Indicates the current count value(BCD) of
                              the High-speed counting module.

● **Driving mode**
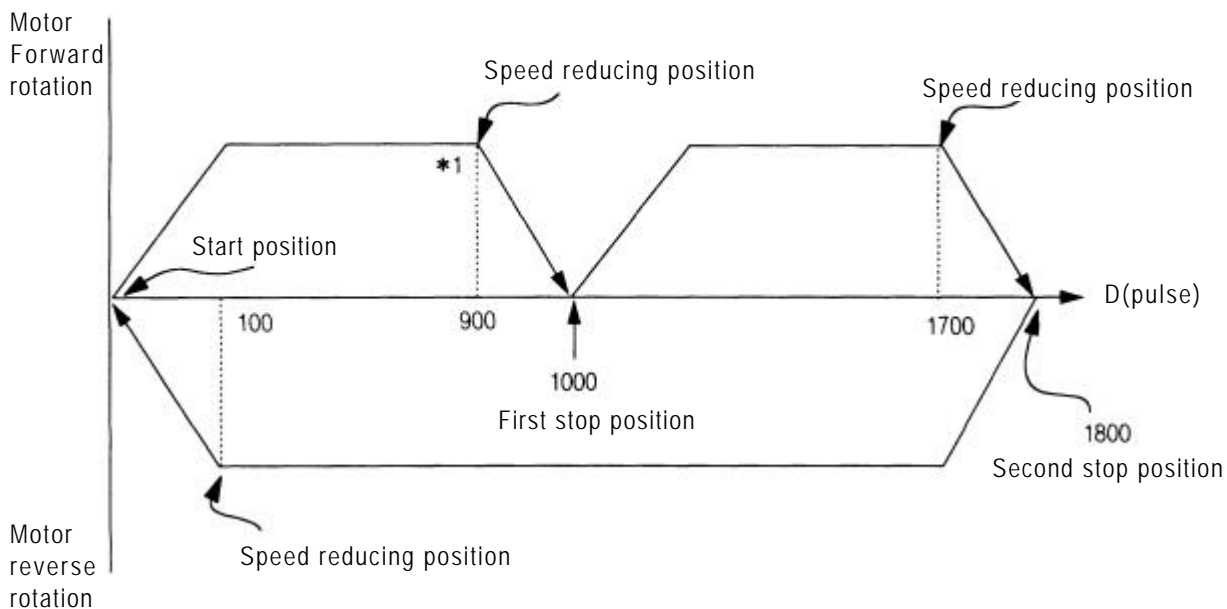
*1. 100(Difference between stop position and speed reducing position) is an interval
       delayed by reducing timing of the inverter.

● **Operation**
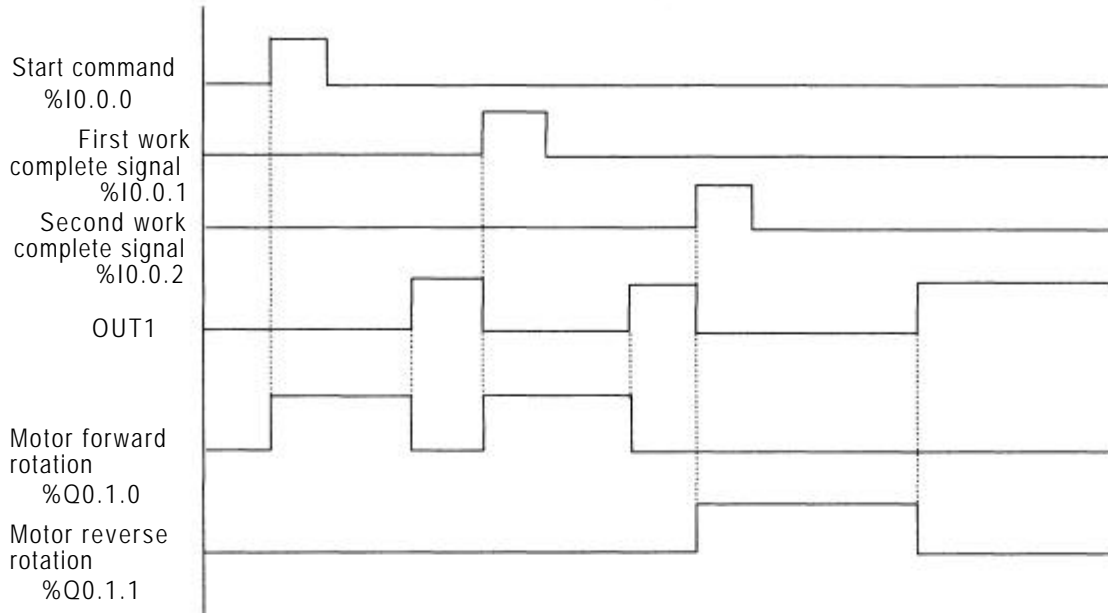
● **Used   Variable**

Variable Name	Var_Kind	Data Type	(AT Address) (Initial Value)
CMP_DATA	: VAR	: UDINT	
CMP_SET	: VAR	: FB Instance	
CMP_SET_DN	: VAR	: BOOL	
CMP_SET_ST	: VAR	: USINT	
CMP1	: VAR	: BYTE	
COUNT_ENABLE	: VAR	: BOOL	
CUR_DATA	: VAR	: UINT	
CUR_VALUE	: VAR	: UDINT	
CY_REST	: VAR	: BOOL	
EQ_RESET	: VAR	: BOOL	
HOME_LATCH	: VAR	: BOOL	
MOVE_ENO	: VAR	: BOOL	
MOVE_ENO1	: VAR	: BOOL	
OUT_EN	: VAR	: BOOL	
OUT1	: VAR	: BOOL	
PRESET	: VAR	: FB Instance	
PRESET_DATA	: VAR	: UDINT	
PRESET_DN	: VAR	: BOOL	
PRESET_ST	: VAR	: USINT	
READ	: VAR	: FB Instance	
READ_DN	: VAR	: BOOL	
UP_DOWN_OUT	: VAR	: BOOL	
WRITE	: VAR	: FB Instance	

● **Program**

Line 0 — %Q0.1.0 OUT1 ———————————— %Q0.1.0 (R)
Line 1 — %Q0.1.1 OUT1 ———————————— %Q0.1.1 (R)
Line 2 — %I0.0.0 P — [MOVE EN ENO] — [MOVE EN ENO] — COUNT_ENABLE (S)
Line 3 — 0—IN1 OUT—PRESET_DATA   900—IN1 OUT—CMP_DATA   %Q0.1.0 (S)
Line 5 — [MOVE EN ENO]
Line 6 — 4—IN1 OUT—CMP1
Line 8 — [PRESET HSC_PRE] REQ DONE—PRESET_DN
Line 9 — 0—BASE STAT—PRESET_ST
Line 10 — 2—SLOT
Line 11 — PRESET_DATA—PSET
Line 14 — %I0.0.1 P — [MOVE EN ENO] — MOVE_ENO ( )   %Q0.1.0 (S)
Line 15 — 1700—IN1 OUT—CMP_DATA
Line 17 — %I0.0.2 P — [MOVE EN ENO] — [MOVE EN ENO] — MOVE_ENO1 ( )   %Q0.1.1 (S)
Line 18 — 100—IN1 OUT—CMP_DATA   1—IN1 OUT—CMP1

Stops forward rotation of the motor when speed-reducing position is over-passed.

Stops reverse rotation of the motor when speed-reducing position is over-passed

If start command turns on, PRESET_DATA = 0 CMP1=4,CMP_DATA=900. COUNT_ENABLE=ON, The motor starts forward

When the first work has been finished, CMP_DATA =1700 and the motor starts forward rotation.

When the second work has been finished CMP_DATA =100 and

5 - 16

```
                                          ┌─CMP_SET─┐
                                          │ HSC_CMP │
           %I0.0.0                         │         │                 When  first  or  second
Line 21  ──┤ P ├──────────┬────────────────┤REQ  DONE├─ CMP_SET_D      works    or    start
                          │                 │         │      N         command   have   been
           MOVE_ENO       │                 │         │                finished,
Line 22  ──┤   ├──────────┤            0 ───┤BASE STAT├─ CMP_SET_S      CMP_DATA    will    be
                          │                 │         │      T
           MOVE_ENO1      │                 │         │
Line 23  ──┤   ├──────────┘            2 ───┤SLOT     │
                                           │         │
Line 24  ──                   CMP_DATA ─────┤CMPD     │
                                           │         │
Line 25  ──                       CMP1 ─────┤CMP1     │
                                           │         │
Line 26  ──                          0 ─────┤CMP2     │
                                           └─────────┘

Line 27  ──

           COUNT_ENA      ┌──WRITE──┐              ┌──READ───┐
           BLE           │ HSC_WR  │              │ HSC_RD  │
Line 28  ──┤   ├──────────┤REQ  DONE├──────────────┤REQ  DONE├─ READ_DN    Write/read        are
                          │         │              │         │            executed   whenever
Line 29  ──          0 ───┤BASE STAT│          0 ──┤BASE STAT│            COUNT_ENABLE = ON
                          │         │              │         │
Line 30  ──          2 ───┤SLOT     │          2 ──┤SLOT  CNT├─ CUR_VALUE
                          │         │              │         │
Line 31  ──       OUT_EN ─┤OT_E     │              │    OUT1├─ OUT1
                          │         │              │         │
Line 32  ──  HOME_LATC ───┤HOME     │              │    OUT2│
                 H        │         │              │         │
Line 33  ──   CY_REST ────┤CY_R     │              │     UP │
                          │         │              │         │
Line 34  ──   EQ_RESET ───┤EQ_R     │              │   HOME │
                          │         │              │         │
Line 35  ── UP_DOWN_O ────┤DOWN     │              │     CY │
                 UT       │         │              │         │
Line 36  ── COUNT_ENA ────┤CT_E     │              │     BW │
                 BLE      └─────────┘              └─────────┘

Line 37  ──

           READ_DN      ┌UDINT_TO_UINT┐          ┌UDINT_TO_BCD┐    The     High     Speed
Line 38  ──┤   ├────────┤EN        ENO├──────────┤EN       ENO├─  Counter        Module
                        │             │          │            │   outputs   the   current
Line 39  ── CUR_VALUE ──┤IN1       OUT├CUR_DATA CUR_DATA─┤IN1  OUT├─ %QW0.1.1  count      value
                        └─────────────┘          └────────────┘   (CUR VALUE) on the
                                                                  digital display.
Line 40  ──
```

5 - 17